```
MMM          MMM  TTTTTTTTTTTTTTT   HHH          HHH  RRRRRRRRRRR     TTTTTTTTTTTTTTT   LLL
MMM          MMM  TTTTTTTTTTTTTTT   HHH          HHH  RRRRRRRRRRRR    TTTTTTTTTTTTTTT   LLL
MMM          MMM  TTTTTTTTTTTTTTT   HHH          HHH  RRRRRRRRRRRR    TTTTTTTTTTTTTTT   LLL
MMMMMM    MMMMMM        TTT         HHH          HHH  RRR         RRR       TTT         LLL
MMMMMM    MMMMMM        TTT         HHH          HHH  RRR         RRR       TTT         LLL
MMMMMM    MMMMMM        TTT         HHH          HHH  RRR         RRR       TTT         LLL
MMM   MMM   MMM         TTT         HHH          HHH  RRR         RRR       TTT         LLL
MMM   MMM   MMM         TTT         HHH          HHH  RRR         RRR       TTT         LLL
MMM   MMM   MMM         TTT         HHH          HHH  RRR         RRR       TTT         LLL
MMM         MMM         TTT         HHHHHHHHHHHHHHHH  RRRRRRRRRRR           TTT         LLL
MMM         MMM         TTT         HHHHHHHHHHHHHHHH  RRRRRRRRRRRR          TTT         LLL
MMM         MMM         TTT         HHHHHHHHHHHHHHHH  RRRRRRRRRRRR          TTT         LLL
MMM         MMM         TTT         HHH          HHH  RRR   RRR             TTT         LLL
MMM         MMM         TTT         HHH          HHH  RRR   RRR             TTT         LLL
MMM         MMM         TTT         HHH          HHH  RRR   RRR             TTT         LLL
MMM         MMM         TTT         HHH          HHH  RRR      RRR          TTT         LLL
MMM         MMM         TTT         HHH          HHH  RRR      RRR          TTT         LLL
MMM         MMM         TTT         HHH          HHH  RRR      RRR          TTT         LLL
MMM         MMM         TTT         HHH          HHH  RRR         RRR       TTT         LLLLLLLLLLLLLLLL
MMM         MMM         TTT         HHH          HHH  RRR         RRR       TTT         LLLLLLLLLLLLLLLL
MMM         MMM         TTT         HHH          HHH  RRR         RRR       TTT         LLLLLLLLLLLLLLLL
```

```
  OOOOOO    TTTTTTTTTT    SSSSSSSS  DDDDDDDD     IIIIII    VV      VV    CCCCCCCC  DDDDDDDD
  OOOOOO    TTTTTTTTTT    SSSSSSSS  DDDDDDDD     IIIIII    VV      VV    CCCCCCCC  DDDDDDDD
OO      OO      TT      SS          DD      DD     II      VV      VV  CC          DD      DD
OO      OO      TT      SS          DD      DD     II      VV      VV  CC          DD      DD
OO      OO      TT      SS          DD      DD     II      VV      VV  CC          DD      DD
OO      OO      TT      SS          DD      DD     II      VV      VV  CC          DD      DD
OO      OO      TT        SSSSSS    DD      DD     II      VV      VV  CC          DD      DD
OO      OO      TT        SSSSSS    DD      DD     II      VV      VV  CC          DD      DD
OO      OO      TT              SS  DD      DD     II      VV      VV  CC          DD      DD
OO      OO      TT              SS  DD      DD     II       VV    VV   CC          DD      DD
OO      OO      TT              SS  DD      DD     II       VV    VV   CC          DD      DD
  OOOOOO        TT      SSSSSSSS    DDDDDDDD     IIIIII      VV      CCCCCCCC  DDDDDDDD
  OOOOOO        TT      SSSSSSSS    DDDDDDDD     IIIIII      VV      CCCCCCCC  DDDDDDDD


LL              IIIIII    SSSSSSSS
LL              IIIIII    SSSSSSSS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II        SSSSSS
LL                II        SSSSSS
LL                II              SS
LL                II              SS
LL                II              SS
LL                II              SS
LLLLLLLLLL      IIIIII    SSSSSSSS
LLLLLLLLLL      IIIIII    SSSSSSSS
```

Table of contents

```
0000     1          .TITLE  OTS$DIVCD - D COMPLEX*16 / D COMPLEX*16 DIVISION ROUTINE
0000     2          .IDENT  /1-001/         ; File: OTSDIVCD.MAR
0000     3
0000     4 ;
0000     5 ;*****************************************************************************
0000     6 ;*                                                                           *
0000     7 ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0000     8 ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     9 ;*  ALL RIGHTS RESERVED.                                                     *
0000    10 ;*                                                                           *
0000    11 ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0000    12 ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    13 ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0000    14 ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0000    15 ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0000    16 ;*  TRANSFERRED.                                                             *
0000    17 ;*                                                                           *
0000    18 ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0000    19 ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000    20 ;*  CORPORATION.                                                             *
0000    21 ;*                                                                           *
0000    22 ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0000    23 ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000    24 ;*                                                                           *
0000    25 ;*                                                                           *
0000    26 ;*****************************************************************************
0000    27 ;
0000    28 ;
0000    29 ; FACILITY: MATH LIBRARY
0000    30 ;++
0000    31 ; ABSTRACT:
0000    32 ;
0000    33 ;       Perform D COMPLEX*16 division
0000    34 ;
0000    35 ;--
0000    36 ;
0000    37 ; AUTHOR:
0000    38 ;       Steven B. Lionel, 12-July-1979
0000    39 ;
0000    40 ; MODIFIED BY:
0000    41 ;
0000    42 ;
0000    43 ;
```

```
0000     45          .SBTTL  HISTORY         ; Detailed Current Edit History
0000     46
0000     47
0000     48 ; Edit History
0000     49 ;
0000     50
0000     51 ; 1-001 - Adapted from OTS$DIVC version 1-003.  SBL 12-July-1979
```

```
                     0000     53          .SBTTL  DECLARATIONS
                     0000     54
                     0000     55 ;
                     0000     56 ; INCLUDE FILES:
                     0000     57 ;
                     0000     58
                     0000     59 ;
                     0000     60 ; EXTERNAL SYMBOLS:
                     0000     61 ;
                     0000     62
                     0000     63 ;
                     0000     64 ; MACROS:
                     0000     65 ;
                     0000     66
                     0000     67 ;
                     0000     68 ; PSECT DECLARATIONS:
                     0000     69 ;
                 00000000     70          .PSECT _OTS$CODE        PIC, USR, CON, REL, LCL, SHR, -
                     0000     71                                  EXE, RD, NOWRT, LONG
                     0000     72
                     0000     73 ;
                     0000     74 ; EQUATED SYMBOLS:
                     0000     75 ;
00000004             0000     76          a       = 4                     ; real part of dividend
0000000C             0000     77          b       = 12                    ; imag part of dividend
00000014             0000     78          c       = 20                    ; real part of divisor
0000001C             0000     79          d       = 28                    ; imag part of divisor
                     0000     80
                     0000     81 ;
                     0000     82 ; OWN STORAGE:
                     0000     83 ;
                     0000     84 ;        none
                     0000     85
```

```
0000     87          .SBTTL  D COMPLEX*16 / D COMPLEX*16 giving D COMPLEX*16 result
0000     88
0000     89 ;++
0000     90 ; FUNCTIONAL DESCRIPTION:
0000     91 ;
0000     92 ;       OTS$DIVCD_R3 - D COMPLEX*16 / D COMPLEX*16 giving D COMPLEX*16 result
0000     93 ;
0000     94 ;
0000     95 ;       The COMPLEX*16 result is computed as follows:
0000     96 ;
0000     97 ;       1)  Let (a, b) represent the COMPLEX*16 dividend.
0000     98 ;       2)  Let (c, d) represent the COMPLEX*16 divisor.
0000     99 ;       3)  Let (r, i) represent the COMPLEX*16 quotient.
0000    100 ;
0000    101 ;       Then:
0000    102 ;
0000    103 ;       r = (ac + bd) / (cc + dd)
0000    104 ;       i = (bc - ad) / (cc + dd)
0000    105 ;
0000    106 ; CALLING SEQUENCE:
0000    107 ;
0000    108 ;       Complex_quotient.wdc.w = OTS$DIVCD_R3(dividend.rdc.v, divisor.rdc.v)
0000    109 ;
0000    110 ; INPUT PARAMETERS:
0000    111 ;
0000    112 ;       Dividend and divisor parameters are represented as
0000    113 ;       FORTRAN D COMPLEX*16 numbers and are CALL BY VALUE.
0000    114 ;       Passing 128 bit quantities by value is a violation
0000    115 ;       of the VAX calling standard, but is excused because
0000    116 ;       this is a code support routine not meant to be
0000    117 ;       callable by users.
0000    118 ;
0000    119 ; IMPLICIT INPUTS:
0000    120 ;       NONE
0000    121 ;
0000    122 ; OUTPUT PARAMETERS:
0000    123 ;       NONE
0000    124 ;
0000    125 ; IMPLICIT OUTPUTS:
0000    126 ;       NONE
0000    127 ;
0000    128 ; FUNCTIONAL VALUE:
0000    129 ;
0000    130 ;       The D COMPLEX*16 value returned is (a, b) / (c, d)
0000    131 ;       in resisters R0-R3!  This is a violation of the VAX
0000    132 ;       calling standard, but is excused because this is
0000    133 ;       a code support routine, not meant to be callable
0000    134 ;       by users.
0000    135 ;
0000    136 ; SIDE EFFECTS:
0000    137 ;
0000    138 ;       Modifies registers R0-R3!
0000    139 ;       SS$_ROPRAND if either argument is a reserved operand.
0000    140 ;       SS$_FLTOVF if floating overflow
0000    141 ;       SS$_FLTDIV if divide by zero
0000    142 ;--
```

```
                                0FF0  0000   144          .ENTRY  OTS$DIVCD_R3, ^M<R4,R5,R6,R7,R8,R9,R10,R11>
                                      0002   145          MTH$FLAG_JACKET                     ; establish math error handler
                                      0002
          6D    00000000'GF     9E    0002               MOVAB   G^MTH$$JACKET_HND, (FP)
                                      0009                                                     ; set handler address to jacket
                                      0009                                                     ; handler
                                      0009
                                      0009   146
                                      0009   147 ; Perform scaling of all operands before division
                                      0009   148 ;
     51   14 AC    08    07    EF    0009   149          EXTZV   #7, #8, c(AP), R1           ; R1 = c(AP)<exp>  0,1,...377
     50   1C AC    08    07    EF    000F   150          EXTZV   #7, #8, d(AP), R0           ; R0 = d(AP)<exp>  0,1,...377
                   51    50    B1    0015   151          CMPW    R0, R1                      ; R0 = MAX (c<exp>, d<exp>)
                         03    14    0018   152          BGTR    2$
                   50    51    B0    001A   153          MOVW    R1, R0
                   50    50    8E    001D   154 2$:      MNEGB   R0, R0                      ; R0 = scaling exponent  0,377,376,...,1
              50   50    07    9C    0020   155          ROTL    #7, R0, R0                  ; build a floating scale factor
                   51    D4    0024   156          CLRL    R1
                                      0026   157                                             ; scale all operands
        58    14 AC    50    65    0026   158          MULD3   R0, c(AP), R8               ; R8-R9 gets c
        5A    1C AC    50    65    002B   159          MULD3   R0, d(AP), R10              ; R10-R11 gets d
        54    04 AC    50    65    0030   160          MULD3   R0, a(AP), R4               ; a
        52    0C AC    50    65    0035   161          MULD3   R0, b(AP), R2               ; b
                                      003A   162
           50    58    54    65    003A   163          MULD3   R4, R8, R0                  ; R0 = ac
           56    5A    52    65    003E   164          MULD3   R2, R10, R6                 ; R7 = bd
                 50    56    60    0042   165          ADDD2   R6, R0                      ; R0 = ac+bd
                 54    5A    64    0045   166          MULD2   R10, R4                     ; R4 = ad
                 52    58    64    0048   167          MULD2   R8, R2                      ; R2 = bc
                 52    54    62    004B   168          SUBD2   R4, R2                      ; R2 = bc - ad
                 58    58    64    004E   169          MULD2   R8, R8                      ; R8 = cc
                 5A    5A    64    0051   170          MULD2   R10, R10                    ; R10 = dd
                 58    5A    60    0054   171          ADDD2   R10, R8                     ; R8 = cc + dd
                 50    58    66    0057   172          DIVD2   R8, R0                      ; R0 = (ac+bd) / (cc+dd)
                 52    58    66    005A   173          DIVD2   R8, R2                      ; R2 = (bc-ad) / (cc+dd)
                                      005D   174
                         04    005D   175          RET                                     ; (R0-R1, R2-R3) = (r, i)
                               005E   176
                               005E   177          .END
```

```
A                 = 00000004
B                 = 0000000C
C                 = 00000014
D                 = 0000001C
MTH$$JACKET_HND   ********    X   01
OTS$DIVCD_R3      00000000 RG   01
```

Psect synopsis

```
PSECT name                      Allocation          PSECT No.  Attributes
----------                      ----------          ---------  ----------
.  ABS  .                       00000000  (    0.)  00 (  0.)  NOPIC   USR   CON   ABS   LCL NOSHR NOEXE NORD  NOWRT NOVEC BYTE
_OTS$CODE                       0000005E  (   94.)  01 (  1.)    PIC   USR   CON   REL   LCL   SHR   EXE   RD  NOWRT NOVEC LONG
```

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.08	00:00:01.21
Command processing	121	00:00:00.65	00:00:04.02
Pass 1	83	00:00:00.67	00:00:03.71
Symbol table sort	0	00:00:00.00	00:00:00.00
Pass 2	47	00:00:00.55	00:00:02.32
Symbol table output	2	00:00:00.01	00:00:00.04
Psect synopsis output	2	00:00:00.02	00:00:00.11
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	286	00:00:01.98	00:00:11.50

```
The working set limit was 900 pages.
3071 bytes (6 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 6 non-local and 1 local symbols.
237 source lines were read in Pass 1, producing 11 object records in Pass 2.
1 page of virtual memory was used to define 1 macro.
```

Macro library statistics

```
Macro library name                          Macros defined
------------------                          --------------
_$255$DUA28:[SYSLIB]STARLET.MLB;2                    0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS$:OTSDIVCD/OBJ=OBJ$:OTSDIVCD MSRC$:MTHJACKET/UPDATE=(ENH$:MTHJACKET)+MSRC
```

OTSMULCD
LIS

OTSPOWCGC
LIS

OTSDIVC
LIS

OTSPOWDD
LIS

OTSPOWCC
LIS

OTSPOWGG
LIS

OTSPOWGJ
LIS

OTSPOWCDJ
LIS

MTHTAN
LIS

MTHVECTOR
LIS

OTSDIVCG
LIS

OTSPOWCJ
LIS

OTSPOWDLJ
LIS

MTHTANH
LIS

OTSMULCG
LIS

OTSPOWCGJ
LIS

OTSPOWDJ
LIS

OTSDIVCD
LIS

OTSPOWCDC
LIS